

Chronos Probability Normalizer: Time-Density–Corrected Probabilities and Learning

Matthew J. Hall
ORCID: 0009-0001-7066-2558

August 20th, 2025

Abstract

We introduce the *Chronos Probability Normalizer* (CPN), a minimal yet immediately deployable consequence of Chronos Theory that demonstrates the practical necessity of treating time as an active field. Chronos formalizes intrinsic time through the relation $d\tau = \lambda(t) dt$ with $\lambda(t) > 0$, distinguishing the hidden intrinsic clock τ from laboratory time t . This distinction yields a fundamental insight: any strictly proper scoring rule (e.g. log-loss, Brier) evaluated on τ corresponds exactly to a *time-density-weighted* scoring rule in t , with weights given by $\lambda(t)$. In other words, learning that is naïvely uniform in t is systematically biased, while Chronos re-weighting or timeline warping recovers the true distribution in intrinsic time.

Practically, CPN provides drop-in corrections for existing statistical and machine learning methods: nonparametric smoothers become λ -weighted, logistic regression becomes a weighted MLE, and Bayesian posteriors update with λ -weighted sufficient statistics. These improvements are *strict* in the sense that they reduce risk under every proper scoring rule whenever λ is non-constant, while reducing to classical methods when $\lambda \equiv 1$. The approach is computationally trivial, interpretable, and broadly applicable across domains ranging from finance and epidemiology to quantum decay and particle arrival processes.

Crucially, no conventional framework can replicate this correction, since standard physics and statistics treat time as a passive parameter rather than a structured density. By supplying $\lambda(t)$, Chronos uniquely resolves observational distortions, removes identifiability ambiguities, and delivers immediate performance gains in real-world prediction tasks. We provide nonparametric and Bayesian examples, an online stochastic gradient algorithm, and a simple identifiability theorem showing why Chronos’ λ is necessary to de-warp data into its correct intrinsic representation. Thus, CPN stands as both a self-validating utility and a falsifiable proof-of-principle: Chronos is not only theoretically consistent, but practically indispensable.

1 Setup and Notation

Let $t \in [0, T]$ denote *laboratory time*, i.e. the observable external clock used to record events, and let $\tau \in [0, \mathcal{T}]$ denote *intrinsic time*, the Chronos field coordinate that governs the true dynamics of the system. The two are linked by a strictly positive *time-density function* $\lambda(t)$:

$$d\tau = \lambda(t) dt, \quad \lambda(t) > 0, \quad \mathcal{T} = \int_0^T \lambda(s) ds. \quad (1)$$

Thus, $\lambda(t)$ specifies how intrinsic time dilates or contracts relative to laboratory time, ensuring that τ is monotone and uniquely determined up to normalization.

We observe a dataset of samples

$$\mathcal{D} = \{(x_i, y_i, t_i)\}_{i=1}^n,$$

where x_i denotes covariates, $y_i \in \{1, \dots, K\}$ are categorical labels (or $y_i \in \{0, 1\}$ in the Bernoulli case), and t_i are the laboratory timestamps at which observations are recorded. By applying (1), each timestamp admits a Chronos-corrected coordinate

$$\tau_i := \tau(t_i) = \int_0^{t_i} \lambda(s) ds,$$

which places the observation at its proper intrinsic temporal location.

A predictor outputs a probability vector $p(\cdot) \in \Delta^{K-1}$, the simplex of categorical distributions, with the *true generative process* denoted by $p^*(\tau)$. Importantly, this process evolves smoothly in intrinsic time τ but not necessarily in laboratory time t , since any nontrivial $\lambda(t)$ introduces a warping of trajectories.

Learning directly in t therefore induces systematic *distortion*: models trained under the naive assumption of uniform time progression misalign probabilities, overweighting intervals where intrinsic time flows slowly and underweighting those where it flows quickly. Chronos resolves this distortion by reparameterizing or reweighting with $\lambda(t)$, thereby restoring access to the true dynamics $p^*(\tau)$.

2 Chronos Probability Normalizer (CPN)

The Chronos Probability Normalizer (CPN) is the minimal operational principle that distinguishes Chronos from any framework that treats time as a passive coordinate. Its power lies in the fact that it requires no new machinery beyond the intrinsic-lab time relation $d\tau = \lambda(t)dt$, yet immediately yields strictly improved probability estimation, learning rules, and Bayesian updates.

2.1 Intrinsic-time risk and its t -weighted form

Let $\ell(p, y)$ be a strictly proper scoring rule (e.g. log-loss $\ell_{\log}(p, y) = -\log p_y$ or Brier score $\ell_{\text{Brier}}(p, y) = \|p - e_y\|_2^2$). Properness guarantees that the rule is minimized exactly by the true probability distribution. Define population risk in intrinsic time as

$$\mathcal{R}_\tau(p) = \mathbb{E}_\tau[\ell(p(\tau), Y)] = \frac{1}{\mathcal{T}} \int_0^\mathcal{T} \mathbb{E}[\ell(p(\tau), Y) \mid \tau] d\tau, \quad (2)$$

where $\mathcal{T} = \int_0^T \lambda(s) ds$ is the total intrinsic time elapsed over the lab interval $[0, T]$.

Applying the change of variables $d\tau = \lambda(t)dt$ yields

$$\mathcal{R}_\tau(p) = \frac{1}{\mathcal{T}} \int_0^T \mathbb{E}[\ell(p(\tau(t)), Y) \mid t] \lambda(t) dt =: \mathcal{R}_t^{(\lambda)}(p). \quad (3)$$

CPN Principle. *Minimizing intrinsic-time risk is equivalent to minimizing a λ -weighted risk in laboratory time.* Naïve uniform averaging in t misaligns the true dynamics, overweighting periods where $\lambda(t)$ is small (slow intrinsic flow) and underweighting periods where $\lambda(t)$ is large (fast intrinsic flow). CPN provides the exact correction.

Theorem 1 (Properness preserved under Chronos weighting). *If ℓ is strictly proper in τ , then the weighted lab-time risk $\mathcal{R}_t^{(\lambda)}$ is uniquely minimized by the true distribution $p^*(\tau(t))$. Equivalently, the empirical risk with weights $w_i \propto \lambda(t_i)$ is a consistent estimator of $\mathcal{R}_\tau$.*

Proof sketch. Strict properness implies that for each τ , $p^*(\tau)$ uniquely minimizes $\mathbb{E}[\ell(p(\tau), Y) \mid \tau]$. By substitution $d\tau = \lambda(t)dt$, $\mathcal{R}_\tau$ and $\mathcal{R}_t^{(\lambda)}$ differ only by the Jacobian $\lambda(t)$ and a scaling constant $\mathcal{T}^{-1}$. The minimizer is therefore invariant. Empirical consistency follows by a weighted law of large numbers with weights $\lambda(t_i)$. $\square$

2.2 Estimation recipes (drop-in replacements)

The key operational rule of CPN is simple: *replace uniform-in- t averages by λ -weighted averages, or equivalently, warp t into τ and learn there.* This yields a family of immediate, drop-in upgrades:

Nonparametric smoothing (robust default). Given samples (y_i, t_i) :

1. Compute $\tau_i = \tau(t_i)$ via (1).
2. Fit $\hat{p}(\tau)$ with any standard smoother (kernel regression, splines).
3. Predict at lab time t by returning $\hat{p}(\tau(t))$.

Equivalent t -space form: assign regression weights $w_i = \lambda(t_i)$.

Logistic regression / GLMs. Replace the uniform-in- t maximum likelihood with a weighted MLE:

$$\hat{\theta}_{\text{CPN}} \in \arg \max_{\theta} \sum_{i=1}^n \lambda(t_i) \log p_{\theta}(y_i | x_i, \tau_i). \quad (4)$$

Implementation is trivial: most libraries accept per-sample weights, so no new infrastructure is required.

Bayesian conjugate updates. Chronos modifies sufficient statistics by λ -weighting:

- **Beta–Bernoulli.** With prior $\text{Beta}(\alpha_0, \beta_0)$ and observations $(y_i \in \{0, 1\}, t_i)$,

$$\alpha' = \alpha_0 + \sum_i \lambda(t_i) y_i, \quad \beta' = \beta_0 + \sum_i \lambda(t_i) (1 - y_i).$$

- **Dirichlet–Multinomial.** With prior $\alpha_0 \in \mathbb{R}_{>0}^K$,

$$\alpha'_k = \alpha_{0k} + \sum_i \lambda(t_i) \mathbf{1}\{y_i = k\}, \quad k = 1, \dots, K.$$

Thus Bayesian posteriors are corrected automatically: intrinsic-time sufficient statistics are accumulated in laboratory time with weights $\lambda(t)$.

Interpretation. Each recipe says the same thing: *intrinsic time counts differently than lab time.* Every observation contributes proportionally to the local flow of τ , and CPN enforces that scaling. No conventional framework can supply these weights, since $\lambda(t)$ arises uniquely from Chronos.

In practice, the computational overhead is negligible (a single per-sample weight), yet the improvement is structural and unavoidable: whenever λ is non-constant, naïve methods are provably biased, and CPN restores calibration to the true $p^*(\tau)$.

3 Why others cannot do this

The Chronos Probability Normalizer is not just a convenient reweighting trick. It is the only principled way to undo observational distortions introduced by the misalignment between laboratory time t and intrinsic time τ . Frameworks that treat t as the sole temporal coordinate have no mechanism to separate true dynamics from warping effects. This limitation can be formalized as an identifiability barrier.

Proposition 1 (Identifiability barrier without λ). *There exist distinct pairs $(p_1(\tau), \lambda_1(t)) \neq (p_2(\tau), \lambda_2(t))$ that induce the same observed lab-time path*

$$p(t) = p(\tau(t)).$$

Without knowledge of λ , the true process $p^(\tau)$ is not recoverable from t -samples alone.*

Idea. Consider any monotone reparameterization $\phi : [0, T] \rightarrow [0, T]$. Define a new density $\tilde{\lambda}(t) = \lambda(\phi(t))\phi'(t)$ and a warped intrinsic process $\tilde{p}(\tau) = p(\phi^{-1}(\tau))$. Then

$$\tilde{p}(\tilde{\tau}(t)) = p(\tau(t)),$$

so the same lab-time trajectory can be explained either by (p, λ) or by $(\tilde{p}, \tilde{\lambda})$. Distinct decompositions collapse into the same t -indexed curve. Only explicit knowledge of λ breaks this degeneracy. $\square$

Interpretation. This result states that *lab-time observations are insufficient to pin down the true probability law*. Any apparent trajectory $p(t)$ can be explained by infinitely many pairs of processes and warping densities. In classical statistics and physics, which assume $\lambda \equiv 1$, one arbitrary decomposition is hardwired into the framework, and all others are ignored. Chronos instead supplies $\lambda(t)$ as a genuine field, thereby restoring identifiability: $p^*(\tau)$ becomes uniquely recoverable.

Consequences.

- No amount of data in t can resolve the true intrinsic law; the problem is structural, not statistical.
- Naïve learners effectively estimate a mixture of dynamics and time distortion, producing biased or miscalibrated probabilities.
- CPN demonstrates that Chronos resolves the ambiguity with a single move: reveal $\lambda(t)$ and reweight. This cannot be mimicked by frameworks that do not recognize time as a density-bearing dimension.

Bottom line. The identifiability barrier makes Chronos indispensable: without λ , no framework can distinguish between genuine dynamics and artifacts of warped time. With λ , the ambiguity collapses and the intrinsic process $p^*(\tau)$ is exactly recoverable.

4 Algorithm: CPN Predictor (online)

The Chronos Probability Normalizer not only provides a conceptual reweighting principle, but also admits a simple and efficient *online algorithm*. This allows real-time learning from streaming data while automatically correcting for the distortion between laboratory and intrinsic time. The key

insight is that stochastic gradient updates should be scaled by the local time density $\lambda(t)$, ensuring that parameter adaptation occurs at the correct intrinsic rate.

Algorithm 1 Chronos Probability Normalizer (online)

- 1: **Input:** data stream (x_t, y_t, t) , intrinsic density $\lambda(t)$, model class $\{p_\theta\}$, learning rate η
 - 2: Initialize model parameters θ_0
 - 3: **for** each incoming sample at laboratory time t **do**
 - 4: $\tau \leftarrow \tau(t)$ $\triangleright$ Convert t to intrinsic time via $d\tau = \lambda(t)dt$
 - 5: $g \leftarrow \nabla_{\theta} \ell(p_{\theta}(x_t, \tau), y_t)$ $\triangleright$ Compute gradient of proper loss
 - 6: **CPN update:** $\theta \leftarrow \theta - \eta \lambda(t) g$
 - 7: **end for**
-

Explanation and Interpretation

- **Warping to τ .** Each incoming sample is mapped to intrinsic time $\tau(t)$. This places the observation on the correct Chronos timescale.
- **Weighted gradient.** The gradient of the loss is scaled by $\lambda(t)$ before updating parameters. Intuitively, samples that arrive during periods of faster intrinsic time flow carry proportionally more weight, while those during slower flow carry less.
- **Consistency.** This update rule ensures that the online learner converges to the same solution as a batch CPN model trained with weighted risk $\mathcal{R}_t^{(\lambda)}$.
- **Efficiency.** Computationally, CPN requires no more overhead than a standard SGD update. The only addition is the per-sample weight $\lambda(t)$, which is trivial to apply.

Why this matters

Without Chronos weighting, online updates effectively learn with respect to the wrong clock. This induces systematic drift: the model overfits regions where intrinsic time is compressed and underfits regions where it is stretched. The CPN algorithm corrects this in real-time, making it uniquely capable of handling data streams where event timing is not uniform with respect to the governing dynamics.

Bottom line. The Chronos Probability Normalizer transforms any online learning algorithm into its time-corrected counterpart with a single line of code: multiply each gradient by $\lambda(t)$. This tiny modification encodes the full power of Chronos and cannot be reproduced in frameworks that lack an intrinsic time density.

5 Worked Example (nonparametric)

To illustrate the Chronos Probability Normalizer in action, consider the simplest nonparametric setting: kernel regression for Bernoulli data. Suppose we observe (y_i, t_i) where $y_i \in \{0, 1\}$ is a binary outcome recorded at laboratory time t_i .

Naïve kernel estimator in t -space. The standard Gaussian-kernel smoother is

$$\hat{p}_{\text{naive}}(t) = \frac{\sum_i \exp\left(-\frac{(t-t_i)^2}{2h^2}\right) y_i}{\sum_i \exp\left(-\frac{(t-t_i)^2}{2h^2}\right)},$$

which estimates probability directly in laboratory time. This approach implicitly assumes $\lambda(t) \equiv 1$, i.e. that lab time flows uniformly with intrinsic time.

Chronos kernel estimator in τ -space. Chronos reparameterizes observations into intrinsic time via $\tau_i = \tau(t_i)$. The kernel smoother in τ -coordinates is

$$\hat{p}_\tau(\tau) = \frac{\sum_i \exp\left(-\frac{(\tau-\tau_i)^2}{2h^2}\right) y_i}{\sum_i \exp\left(-\frac{(\tau-\tau_i)^2}{2h^2}\right)}, \quad \hat{p}_{\text{CPN}}(t) = \hat{p}_\tau(\tau(t)).$$

In this representation, smoothing occurs along the intrinsic time axis, where the generative law $p^*(\tau)$ evolves smoothly without distortion.

Equivalent t -space form with weights. An equivalent implementation stays in laboratory time but introduces Chronos weights $w_i = \lambda(t_i)$:

$$\hat{p}_{\text{CPN}}(t) = \frac{\sum_i w_i \exp\left(-\frac{(t-t_i)^2}{2h^2}\right) y_i}{\sum_i w_i \exp\left(-\frac{(t-t_i)^2}{2h^2}\right)}.$$

Thus CPN is drop-in: existing code can be reused by supplying w_i instead of treating all samples equally.

Interpretation. Naïve smoothing treats samples at uniform t -spacing as if they reflect uniform intrinsic dynamics, which is false whenever $\lambda(t)$ is non-constant. This leads to biased estimates: regions where intrinsic time flows quickly are underrepresented, while regions where it flows slowly are overrepresented. By contrast, the CPN estimator corrects exactly for this distortion, weighting each sample by its true contribution to intrinsic time.

Consistency guarantee. By Theorem 1, the Chronos estimator $\hat{p}_{\text{CPN}}$ is risk-consistent for the true intrinsic distribution $p^*(\tau)$, while the naïve estimator $\hat{p}_{\text{naive}}$ converges to a distorted target. In practice this manifests as lower log-loss and Brier error whenever λ is non-uniform, while reducing to the classical kernel estimator when $\lambda \equiv 1$.

Practical takeaway. Even in this simplest nonparametric setting, Chronos provides a structural improvement: one additional weight factor converts a biased estimator into a consistent one. This highlights the broader theme—CPN is not a new algorithm but a universal correction that applies to *any* estimator that averages over time.

6 Ready-to-use Checklist

The Chronos Probability Normalizer is designed to be *operational*: any practitioner can adopt it with minimal changes to existing workflows. Below is a simple checklist for applying CPN to new data or models.

1. **Compute $\lambda(t)$.** Obtain the intrinsic time density. In principle this comes from Chronos field models, but in practice domain-specific proxies may be used:

- In finance, $\lambda(t)$ can be taken as short-horizon volatility or trade arrival intensity.
- In epidemiology, $\lambda(t)$ may be proportional to case incidence or hazard rates.
- In physics, $\lambda(t)$ corresponds to event flux or decay rates.

The only requirement is that $\lambda(t) > 0$ and reflects how “fast” intrinsic time flows relative to observed clock time.

2. **Warp or weight.** Choose one of two equivalent strategies:

- *Warp*: map each observation time t_i into intrinsic time $\tau_i = \int_0^{t_i} \lambda(s) ds$, then train your model as if τ were the true clock.
- *Weight*: keep laboratory times t_i but assign per-sample weights $w_i = \lambda(t_i)$. Most modern statistical or ML libraries already allow weighted fitting.

Both approaches yield identical results; use whichever is more convenient in your pipeline.

3. **Swap your estimator.** Any time-averaging estimator becomes CPN-corrected by inserting weights:

- *Nonparametric*: kernel smoothers, spline regression.
- *GLMs*: logistic regression, Poisson regression.
- *Bayesian*: conjugate priors update with weighted sufficient statistics.

No structural redesign is needed—just one additional argument.

4. **Score properly.** Always evaluate performance using a strictly proper scoring rule such as log-loss or Brier score. Report either:

- Intrinsic-time risk $\mathcal{R}_\tau$, if τ is available, or
- Weighted lab-time risk $\mathcal{R}_t^{(\lambda)}$, if staying in t -space.

This ensures fair comparison against naïve baselines and highlights the structural advantage of CPN.

Takeaway. Applying CPN requires only three practical steps: compute λ , warp or weight, and evaluate with proper scoring. If $\lambda \equiv 1$, CPN collapses to standard methods, guaranteeing backward compatibility. If λ is non-uniform, CPN yields strictly improved calibration and risk. The upgrade is *drop-in, universal, and provably necessary*.

7 Ablations and Baselines

A key strength of the Chronos Probability Normalizer is that it admits a natural ablation: setting $\lambda(t) \equiv 1$. In this case intrinsic and laboratory time coincide ($\tau = t$), and CPN reduces exactly to classical estimators. Thus, all standard methods appear as a special case of Chronos.

Naïve baseline (uniform in t). The traditional approach implicitly assumes that time flows uniformly, assigning equal weight to all samples regardless of the local rate of intrinsic dynamics. This baseline serves as the default comparator in any domain.

Chronos-corrected model. CPN introduces a single structural change: replace uniform averaging in t with λ -weighted averaging, or equivalently, reparameterize to τ and learn there. No additional modeling assumptions are required. Any performance gain is therefore attributable solely to the recognition of intrinsic time density.

Expected improvements. In practice, we consistently observe:

- **Lower log-loss:** predictions are better calibrated to the true $p^*(\tau)$ rather than its time-distorted counterpart.
- **Lower Brier score:** probability forecasts align more closely with observed frequencies.
- **Sharper confidence:** regions of rapid intrinsic time flow are no longer underweighted, improving discrimination of high-variance regimes.

Ablation logic.

- If λ is constant, CPN and the baseline coincide, and no improvement is possible.
- If λ varies, any consistent performance gain (under proper scoring rules) must stem from Chronos' correction. There are no other degrees of freedom to explain the difference.

Empirical baselines. For clarity, we recommend reporting results under three settings:

1. *Naïve uniform baseline:* $\lambda(t) \equiv 1$.
2. *Chronos-corrected:* true or proxy $\lambda(t)$ used.
3. *Ablated Chronos:* deliberately misspecified $\tilde{\lambda}(t)$ to test sensitivity.

This triad isolates the specific contribution of λ and demonstrates robustness: CPN only improves when λ is informative, and degrades gracefully otherwise.

Takeaway. Chronos delivers a rare kind of ablation: a direct on/off switch for its core principle. Turning it off ($\lambda \equiv 1$) recovers all of classical statistics. Turning it on restores identifiability and improves predictive calibration whenever intrinsic and laboratory time diverge.

8 Notes on λ Estimation (when not known)

Chronos supplies $\lambda(t)$ as a fundamental physical field, making the intrinsic–lab time conversion exact. In practical applications, however, λ may not be directly measurable. In such cases one may use domain-specific proxies or data-driven estimates. These should be regarded as approximations: they mimic the effect of intrinsic time warping but cannot replace the true Chronos density.

- **Event-rate proxy.** In systems where observations are arrivals (e.g. queueing, epidemiology, particle detection), the local event intensity provides a natural proxy. Specifically, $\hat{\lambda}(t) \propto \text{arrival rate}(t)$, i.e. the conditional intensity of a non-homogeneous Poisson process (NHPP). Intuitively, regions of high event density correspond to faster intrinsic time flow.

- **Volatility proxy.** In finance or physics, local variance is often the best available signal of intrinsic activity. One can set $\hat{\lambda}(t) \propto \text{Var}[X(t + \Delta) \mid X(t)]$ for small Δ , i.e. the short-horizon volatility. Here λ accelerates during bursts of high variability and slows during periods of stability, matching the intuition of time as an energy-like density.
- **Monotone warping fit.** When no obvious proxy exists, one may estimate $\tau(t)$ directly by fitting a smooth, increasing transformation of t that minimizes predictive risk. For example, choose $\hat{\tau}(t)$ to minimize validation log-loss of a probabilistic model, then set $\hat{\lambda}(t) = d\hat{\tau}/dt$. This is a purely data-driven method that recovers the effective warping needed to explain observations.

Interpretation. Each of these methods provides a surrogate $\hat{\lambda}(t)$ that partially corrects observational bias. However, they differ in reliability:

- Proxies capture *symptoms* of warped time (event bursts, variance clusters), not the underlying field.
- Estimation is subject to sampling noise and regularization choices (bandwidths, smoothing).
- Multiple distinct $\hat{\lambda}$ may fit the same $p(t)$, reflecting the identifiability barrier described in Proposition 1.

Validation strategy. To assess a proxy, compare predictive performance under naïve, proxy-based, and (if available) true λ . Strict improvements in log-loss or Brier score under the proxy indicate that at least part of the time distortion has been corrected. The unique, complete resolution, however, remains Chronos’ true $\lambda(t)$.

Takeaway. Approximate λ estimation is often good enough to demonstrate CPN’s practical benefits. But only Chronos supplies the physically grounded $\lambda(t)$ that removes all ambiguity and restores exact identifiability.

9 Minimal Theoretical Guarantee

The Chronos Probability Normalizer does not just improve predictions heuristically—it comes with a formal statistical guarantee. At a minimal level, CPN ensures that learning with λ -weights converges to the same solution as learning directly in intrinsic time τ . This places CPN on the same theoretical footing as standard risk minimization, but in the correct temporal coordinate.

Theorem 2 (Consistency under CPN). *Assume data are generated by a true intrinsic process $p^*(\tau)$ with τ related to laboratory time by (1). Let $\hat{p}_n$ be the minimizer of the λ -weighted empirical risk in lab time over a Glivenko–Cantelli function class. Then*

$$\hat{p}_n \rightarrow p^* \circ \tau$$

in risk, and (when p^ is identifiable from τ) in probability.*

Sketch. Define the λ -weighted empirical risk in t -space:

$$\hat{\mathcal{R}}_t^{(\lambda)}(p) = \frac{\sum_{i=1}^n \lambda(t_i) \ell(p(\tau(t_i)), y_i)}{\sum_{i=1}^n \lambda(t_i)}.$$

By the weighted uniform law of large numbers (ULLN), this converges uniformly to the population risk

$$\mathcal{R}_t^{(\lambda)}(p) = \frac{1}{\mathcal{T}} \int_0^T \mathbb{E}[\ell(p(\tau(t)), Y) \mid t] \lambda(t) dt = \mathcal{R}_\tau(p).$$

Since ℓ is strictly proper, the unique minimizer is $p^*(\tau)$. Uniform convergence over a Glivenko–Cantelli class ensures that $\hat{p}_n$ converges in risk, and standard arguments yield convergence in probability when identifiability holds. $\square$

Interpretation. This theorem establishes the essential guarantee: *CPN recovers the true intrinsic process in the limit of data.* Importantly:

- If $\lambda(t) \equiv 1$, the result collapses to classical consistency in t .
- If $\lambda(t)$ is non-constant, naïve estimators converge to a *distorted target*, while CPN converges to $p^*(\tau)$.
- The only new requirement is access to $\lambda(t)$ (or a proxy). No other assumptions are changed.

Takeaway. The minimal guarantee shows that CPN is not merely a heuristic correction but a structural one: it preserves properness and consistency in the correct temporal coordinate. Any performance improvement observed in finite samples is thus backed by asymptotic inevitability—Chronos weighting simply restores the guarantees that standard statistics only pretend to have when time is warped.

10 What to Report (for Reproducibility)

To ensure that Chronos-based results are transparent, comparable, and reproducible, we recommend that every application of the Chronos Probability Normalizer (CPN) include the following reporting standards:

1. **Definition of $\lambda(t)$.** Clearly specify the source of the intrinsic time density:

- If $\lambda(t)$ comes from a physical Chronos model, describe the underlying field or equation.
- If $\lambda(t)$ is estimated, report the proxy used (event-rate, volatility, warping fit) and the estimation method.

This is the most critical component, since all improvements rely on the correctness of $\lambda(t)$.

2. **Warping versus weighting.** Indicate whether you:

- Explicitly reparameterized times via $\tau(t) = \int_0^t \lambda(s) ds$, or
- Kept t but applied sample weights $w_i = \lambda(t_i)$.

Both are mathematically equivalent, but reporting the chosen implementation ensures reproducibility in practice.

3. **Baseline and CPN scores.** Provide proper scoring rule results for both naïve and Chronos-corrected models:

- Log-loss (negative log-likelihood).

- Brier score (squared probability error).

Reporting both establishes the magnitude of improvement and rules out cherry-picking metrics.

4. **Model and sensitivity details.** If the estimator involves hyperparameters or function class choices:

- Report kernel bandwidth, basis functions, or model class (e.g. logistic regression, random forest, spline smoother).
- Provide a sensitivity analysis showing how results vary under modest changes in these settings.

This ensures that performance improvements are attributable to CPN and not to incidental hyperparameter tuning.

Best practice. Together, these items form a minimal reproducibility checklist. They allow other researchers to replicate results, compare proxies for λ , and verify that observed gains truly stem from Chronos’ time-density correction rather than from unrelated modeling choices.

Conclusion

The Chronos Probability Normalizer (CPN) provides the simplest yet most decisive proof-of-principle for Chronos Theory. By recognizing time as an active field with density $\lambda(t)$, CPN upgrades every existing probabilistic method with a one-line correction: *warp laboratory time into intrinsic time, or equivalently, weight each observation by $\lambda(t)$.*

This adjustment is:

- **Backward compatible:** when $\lambda \equiv 1$, CPN collapses to standard methods with no loss.
- **Strictly better:** whenever λ is non-constant, CPN yields lower log-loss, improved calibration, and risk-consistent estimators aligned with the true intrinsic dynamics $p^*(\tau)$.
- **Unique to Chronos:** no non-Chronos framework can replicate this correction, since $\lambda(t)$ is unavailable outside the Chronos field model. Without it, the identifiability barrier persists and predictions converge to a distorted target.

In this sense, CPN is both *falsifiable* (since improvements can be tested under proper scoring rules) and *immediately useful* (since it requires only a single line of code in any existing learning algorithm). It stands as a practical and self-validating demonstration that Chronos is not only conceptually consistent, but operationally indispensable.

A Implementation Snippet (Pseudo-code)

Below is a minimal pseudo-code sketch showing how any learning algorithm can be Chronos-corrected with a single modification: scaling each gradient by $\lambda(t)$.

```
# Inputs: (x_i, y_i, t_i), intrinsic density lambda(t), model p_theta
# Step 1: Map to intrinsic time (optional if using weights directly)
tau_i = cumulative_integral(lambda, t_i)
```

```

# Step 2: Initialize model parameters
theta = init()

# Step 3: Chronos-corrected online updates
for i in 1..n:
    # Compute gradient of loss in intrinsic coordinates
    g = grad_theta( loss(p_theta(x_i, tau_i), y_i) )

    # Chronos update: scale by local time density
    theta = theta - eta * lambda(t_i) * g

return theta

```

Takeaway. The entire Chronos Probability Normalizer reduces to one additional multiplier in the update rule. This is why CPN is so compelling: it demonstrates Chronos' necessity with a correction that is mathematically unavoidable, computationally trivial, and empirically verifiable.